Two-Round Multi-Facility Discrete Voronoi Game on a Line

Hamid Zarrabi-Zadeh

Department of Computer Engineering, Sharif University of Technology, Tehran, Iran.
zarrabi@sharif.edu

Abstract. In this paper, we consider a multi-facility version of the two-round discrete Voronoi game on a line. The game consists of two players, and a finite set of users placed along a line. Each player has $2m$ facilities, where $m \geqslant 1$ is a fixed integer. The game starts by the first player, placing 2 facilities on the line, followed by which the second player places 2 facilities, and this continues for multiple rounds. Assuming that each user is served by a facility closest to it, the payoff of each player is defined as the number of users served by the facilities of that player. The goal of each player is to maximize his payoff. In this paper, we present optimal strategies for last move of both players to find their best placements in the game. This is the first result of this kind for the Voronoi game in discrete setting. At the end of paper, we present a new version of this game called, blined version, which the players do not know the distribution of users.

1 Introduction

In a facility location problem, we are concerned with finding a placement for a set of facilities that meet certain optimality criteria. A competitive facility location problem consists of a set of players who are faced with the problem of placing their facilities, in a certain order, that maximize a payoff function. Each facility location problem has a set of users, who require some kind of service. In order to get this service every user goes to its nearest facility, with respect to an appropriate distance measure. On the other hand, every facility has a service zone which is the set of users that are served by it. To be precise, given a set of facilities F the service zone of a facility $f \in F$, denoted by $U(f, F)$, is the set of points in the user space that are closer to f than to any other facility $f' \in F$.

Voronoi game is a two player variation of the competitive facility location problem. Every instance of the game consists of a set of users, U, on a plane. Every user seeks to be served by the closest facility on the plane, regardless of who placed it there. P1 and P2 compete over the users by alternatively placing facilities on the plane, each one trying to maximize the number of users they serve.

Different variations of the Voronoi game exists. In a one-round Voronoi game each player plays once, namely, P1 places all its m facilities after which P2 places the same number of facilities on the plane. In an k-round Voronoi game,

players place one facility at a time, alternatively, for k rounds. The users can be thought of to be either discrete or continuous. Discrete means that the users are represented by points on the plain, and continuous means that the users are distributed inside a region.

The term Voronoi game was first introduced by Ahn *et al.* [1]. They have considered the game where the users are distributed uniformly on a unit length line segment. Player 1 and Player 2 will alternatively place a set of m facilities, F and S respectively, on the line segment. The payoff of P2 is computed by the following function:

$$\mathcal{P}_2(F,S) = |\bigcup_{s \in S} U(S, F \cup S)| \tag{1}$$

The payoff of P1 will be easily computed by $\mathcal{P}_1(F,S) = |U| - \mathcal{P}_2(F,S)$. With respect to these constraints, Ahn *et al.* [1] showed that P2 always has a winning strategy that guarantees a proportional payoff of $1/2 + \epsilon$, with $\epsilon > 0$. However, P1 can force ϵ to be arbitrarily small. On the other hand, in the one-round Voronoi game with m facilities, the first player always has a winning strategy.

In $\mathbb{R}^2$, Cheong *et al.* [9] considered the Voronoi game for a square-shaped user region. They proved for a large enough m, in a one-round game with m facilities, there is always a placement of player 2 that guarantees a payoff of at least $1/2 + \alpha$, where $\alpha > 0$. Fekete and Meijer [11] completed Cheong *et al.*'s work by studying the one-round game played on a rectangular user region with aspect ratio ρ. The Voronoi game played on the edges and vertices of a graph was considered in a paper by Bandyapadhyay *et al.* [2].

Given a set of facilities currently placed on the plane by P1, the problem of finding a placement for player 2 to place one new facility and achieve maximum payoff was considered by Cabello *et al.* [8] and is referred to as the $MaxCov$ problem. This was the first algorithm to be presented for the discrete user space. Later, Bhattacharya and Nandy [7] studied the 2-$MaxCov$ problem, which considers the problem of placing two new facilities by the second player. Before all of this, Dürr and Thang [10] had established an NP-hardness result for deciding the existence of a Nash equilibrium in a given graph. Teramoto *et al.* [12] studied the same problem and considered a much more restricted problem. The first player occupies one vertex of a given graph and the second player owns m vertices. They proved in this case deciding whether the second player has a winning strategy or not is NP-hard.

Recently, Banik *et al.* [3] studied the discrete one-round Voronoi game in $\mathbb{R}$, and introduced algorithms for finding both players' optimal strategies. The authors showed finding the optimal placement for the first player can be done in linear time and the optimal placement for the second player can also be found in $O(n^{m-\lambda_m})$, where m is the number of facilities and $0 < \lambda_m < 1$ depends only on m. After this result, Banik *et al.* [4] studied the two-round Voronoi game and gave algorithms for finding the optimal placements of all the moves. They also showed P2 always has a winning strategy in such a game. The discrete Voronoi game in a polygonal domain was studied in [6] by the same author. In [5] Banik *et al.*studied another variation of the game in $\mathbb{R}^2$. They considered the one-round

discrete Voronoi game in $\mathbb{R}^2$ where both players have already placed k facilities on the plane. In a sense they tried to solve the last round of a $(k+1)$-round game. The authors provided algorithms for both of the players to find their optimal placements.

Our Results. In this paper, we consider last round of the Voronoi game in $\mathbb{R}$, where each player is allowed to place 2 facilities in a round. We call this problem multi-facility Voronoi game. To the extent of our knowledge, this problem has never been studied before in the discrete user space model.

Let the game be two round. The game started by the first player, who has placed f_1 and f_2. The second player has followed this move by placing s_1 and s_2. In the last round the first player places f_3 and f_4 followed by which the second player places s_3 and s_4. Like before, we have a finite set of n users U, where every user goes to its nearest facility to be served. Likewise, every facility serves a set of users. If we denote all facilities by H and let h be an arbitrary facility, $U(h, H)$ is the set of users that are served by h. Consider F and S to be the set of facilities owned by P1 and P2 respectively. The payoff of the second player can be computed using Equation (1) and the payoff of the first player is the difference of the number of users and P2. P1's goal is to find the placements f_1 through f_4, namely F, that minimizes the following function.

$$\max_{s_1,s_2,s_3,s_4} \mathcal{P}_2(F, \{s_1, s_2, s_3, s_4\})$$

P2 also likes to find an optimal placement that minimizes P1's maximum payoff.

We construct an instance of the game in which neither of the players have a winning strategy and in the best case the game ends in a tie. This lack of winning strategy gives an extra importance to the algorithms that we introduced.

For each round of the game, we found a set of placements for each player with the property that the changes in the payoff happens when the player places its facilities on those points. The set of mentioned placements for the last round have a cardinality of $O(n)$ and $O(n^2)$ for P2 and P1, rescpectivly.

The first question to answer in any competitive game is whether any of the players has a winning strategy. For the continuous setting on a line, Ahn *et al.* [1] showed that the second player always has a winning strategy. In the discrete setting, however, none of the players has a winning strategy, as stated in an example in Section A.2.

2 Optimal Strategy for the Last Move of Player 2

The last move of P2 involves finding two locations x and x' for the last two facilities of P2 in order to maximize the payoff

$$\mathcal{P}_2(\{f_1, f_2, f_3, f_4\}, \{s_1, s_2, x, x'\}),$$

provided that the other six facilities, f_1, f_2, f_3, f_4, s_1, and s_2 are given.

We define a *facility interval* to be a maximal interval whose interior is free of any facility. According to this definition, a facility interval can only contain facilities on its endpoints. If an endpoint p of an interval coincides with a facility owned by a player, then we say that the endpoint p is owned by that player. A facility interval is *owned by* a player, if both its endpoints are owned by that player. Otherwise, the facility interval is called a *shared interval.* There are two possible cases for a facility interval to be shared: when the endpoints of the interval are owned by different players, and when the interval has at most one endpoint (i.e., is unbounded). We prove two simple observations regarding facility intervals.

Observation 1 *Let I be a shared facility interval. Then P2 can serve all users in I by placing a single facility.*

Proof. P2 can place its new facility next to the facility owned by P1.

Observation 2 *Let $I = [a, b]$ be a facility interval owned by $P1$. Then P2 can serve all users in a subinterval of I of length $\frac{1}{2}(b-a)$ using only one facility. Using two facilities, P2 can serve all users in I.*

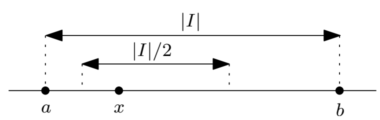

Fig. 1. A subinterval of I covered by placing a single facility at location x.

Proof. If P2 places a single facility at location $x \in [a, b]$, then all users in the subinterval $[\frac{x-a}{2}, \frac{b-x}{2}]$ will be served by x (see Figure 1). The length of this subinterval is $\frac{1}{2}[(b-x)+(x-a)]$. Moreover, P2 can serve all facilities in $[a, b]$ by placing two facilities next to a and b.

Let $N[a, b]$ denote the number of users in the interval $[a, b]$. In light of Observation 2, for a facility interval $[a, b]$, we define:

$$\operatorname{cov}(a, b) = \max_{x \in (a,(a+b)/2)} N[x, x + \frac{1}{2}(b-a)].$$

Note that if a and b are owned by P1, then $\operatorname{cov}(a, b)$ is the maximum number of users that P2 can gain by placing one facility in (a, b). The following lemma is a direct corollary of [3].

Lemma 1 ([3]). *For any facility interval $[a, b]$, $\operatorname{cov}(a, b)$ can be computed in $O(N(a, b))$ time.*

Observation 3 *For any facility interval $[a, b]$, $\operatorname{cov}(a, b) \geqslant N(a, b)/2$.*

Proof. The two subintervals $(a, (a+b)/2]$ and $[(a+b)/2, b)$ together cover all users inside the interval $[a, b]$. By Observation 2, using a single facility, P2 can cover at least one of these two subintervals, one of which contains at least $N(a, b)/2$ users.

Now, we have all ingredients to state the following theorem.

Theorem 4. *Given the facilities f_1, f_2, f_3, f_4, s_1, and s_2, the best placement of s_3 and s_4 for P2 can be computed in $O(n)$ time.*

Proof. The algorithm is as follows. For each facility interval, we compute the maximum payoff of placing a single facility in $O(n)$ time, using Observations 1 and 2, and Lemma 1, and put the computed values in a list. Then, for every facility interval $[a, b]$ owned by P1, we add the value $N(a, b) - \mathrm{cov}(a, b)$ to the list. We then return the two placement with the maximum values in the list. To show that the algorithm is correct, consider an optimal solution OPT, and let s_1 and s_2 be the location of the last two facilities in OPT. If s_1 and s_2 lie in different facility intervals, then the maximum payoff from these two placements exist in the list, and hence, the solution returned by our algorithm is at least as good as the optimal one. If s_1 and s_2 lie in the same facility interval $I = [a, b]$ owned by P1, then the maximum payoff obtained for P2 is $N(a, b)$. However, we have both $\mathrm{cov}(a, b)$ and $N(a, b) - \mathrm{cov}(a, b)$ in the list, and therefore, the solution returned by our algorithm is at least $\mathrm{cov}(a, b) + N(a, b) - \mathrm{cov}(a, b) = N(a, b)$. Note that, by Observation 3, when $N(a, b) - \mathrm{cov}(a, b)$ is one of the maximum values in the list, then the other maximum is $\mathrm{cov}(a, b)$.

Corollary 5 *If P2 can place more than two facilities in last round. His placement can be computed in $O(n)$.*

Proof. Consider the list from last theorem. If P2 can place k facilities, we return k placement with maximum values in the list.

3 Optimal Strategy for the Last Move of Player 1

In this section, we show how to find the placements x and x' for the first player that minimizes

$$\max_{y, y' \in \mathbb{R}} \mathcal{P}_2(\{f_1, f_2, x, x'\}, \{s_1, s_2, y, y'\}),$$

given the locations of the facilities f_1, f_2, s_1, and s_2.

Definition 1. *The set S is* ***sufficient*** *if for each two points like x and y, there are $f_3, f_4 \in S$ such that $P_{opt}(x, y) \leqslant P_{opt}(f_3, f_4)$ which $P_{opt}(a, b)$ is an P1's optimal payoff using a and b as facilities.*

Definition 2. *e is a small value which we discuss about its properties, and we compute it later. x is an* ***event point*** *if at least one of following conditions hold:*

1. *$x \in U$. U is a set which contains all of users.*

2. *If there is a user exactly in the middle of x and a facility of P2, such that there is not any facility between these two points. We define a set which its members have this condition.*

$$\mathcal{D} = \{x \mid \text{ } x \textit{ be in condition 2}\}$$

3. *If $\mathrm{cov}(f, x) < \mathrm{cov}(f, x + e)$ which f is a facility of P1 on the left side of x, and there is not any facilities between x and f.*
4. *If $\mathrm{cov}(x, f) > \mathrm{cov}(x - e, f)$ which f is a facility of P1 on the right side of x, and there is not any facilities between x and f.*

We define a set which its members have condition 3 or condition 4.

$$\mathcal{W} = \{x \mid \text{ } x \textit{ be in condition 3 or condition 4}\}$$

We show event point set with $\mathcal{E}$.

Lemma 2. *The set $\mathcal{E}$ has $O(n)$ members.*

Proof. It is trivial that the cardinality of U and $\mathcal{D}$ is $O(n)$. We show that number of members with condition 3 in $\mathcal{E}$ is $O(n)$. The proof for condition 4 is similar. Let FI be a facility interval which its left end point is P1's facility, like f. According to condition 3 of Definition 2, we know there is an event point if $\mathrm{cov}(f, x)$ to be changed. Cover is a monotone function, which means for $x > y$ we have $\mathrm{cov}(f, x) \geqslant \mathrm{cov}(f, y)$. Also, cover has at most n distinct values; thus, the number of event points in condition 3 is $O(n)$.

Definition 3. *We define point set*

$$\mathcal{W}' = \{x + e \mid x \textit{ be in condition 3 of event points definition}\}$$

$$\cup \{x - e \mid x \textit{ be in condition 4 of event points definition}\}$$

Actually, $\mathcal{W}'$ contains points which value of *cov* changes. These points are computable, and using these points and value of e, we can obtain members of $\mathcal{W}$.

Definition 4. *The aforementioned e has following features:*

1. *If x follows condition 3 of event points definition, then there is no event point and a member of $\mathcal{W}'$ in the interval of $[x, x + e]$.*
2. *If x follows condition 4 of event points definition, then there is no event point and a member of $\mathcal{W}'$ in the interval of $[x - e, x]$.*
3. *If two point like $x, y \in U \cup \mathcal{D} \cup \mathcal{W}'$ is in the same facility interval and $x < y$, then $\mathrm{cov}(x - e, y + e) = \mathrm{cov}(x, y)$, and $N(x - e, y + e) = N(x, y)$.*

Lemma 3. *Let (a, b) and (c, d) be two intervals such that $(a, b) \subset (c, d)$ and there is not any facility in (c, d). Then $\mathrm{cov}(a, b) \leqslant \mathrm{cov}(c, d)$.*

Proof. Let $(x, y) \subset (a, b)$ be an interval such that $N(x, y) = \mathrm{cov}(a, b)$ and $y - x \leqslant (b - a)/2$. According to $(a, b) \subset (c, d)$, we have $(x, y) \subset (c, d)$ and $y - x \leqslant (d - c)/2$. Therefore, from Observation 2 we have $\mathrm{cov}(a, b) = N(x, y) \leqslant \mathrm{cov}(c, d)$.

Theorem 6. *The set of event points is a **sufficient** set.*

There are disntict cases to prove the above theorem. To see the proof and scrutinize on that see Section A.1.

Corollary 7 *There is an optimal placement of facilities f_3 and f_4 such that $f_3, f_4 \in \mathcal{E}$.*

Proof. Based on Theorem 6 this lemma is trivial.

Based on Corollary 7, we can find optimal placement by considering points in $\mathcal{E}$. For this purpose, it is necessary to find members of U, $\mathcal{D}$, and $\mathcal{W}'$. Finding U and $\mathcal{D}$'s members in $O(n)$ is straightforward. In the following, we present an algorithm to find $\mathcal{W}'$ members; afterwards, we provide another algorithm to find the value of e. Then, we can find $\mathcal{W}$'s members using $\mathcal{W}'$ members and e value.

Definition 5. *Consider a facility interval which the left endpoint of this interval is LE, and p is an arbitrary point in this interval. Consider v to be a user inside (LE, p). Let $q = min(p, \frac{p-LE}{2} + v_i)$. We define $cov_v(LE, p) = |U \cap N[v, q]|$.*

We use above definition to find $\mathrm{cov}(LE, p)$.

Lemma 4. *Consider a facility interval which the left endpoint of this interval is LE, and p is an arbitrary point in this interval. We have:*

$$\mathrm{cov}(LE, p) = \max_{u \in U \cap (LE, p)} cov_u(LE, p)$$

Proof. Consider $\mathrm{cov}(f, p) = N[a, b]$ which means $[a, b]$ is the interval corresponding to $\mathrm{cov}(f, p)$. Let a' be closest user on the right side of a. We define $b' = min(p, a' + b - a)$. $N[a', b'] \ngtr N[a, b]$ since $[a, b]$ is the interval corresponding to $\mathrm{cov}(f, p)$. Furthermore, $N[a', b'] \nless N[a, b]$ because $[a', b']$ has all the users of $[a, b]$. Thus, $N[a', b'] = N[a, b] = \mathrm{cov}(f, p)$. In addition, we have $N[a', b'] = cov_{a'}(f, p)$ based on Definition 5. So, the proof is complete.

Lemma 5. *There is an $O(n^2 \log n)$ time algorithm that can compute the set $\mathcal{W}'$.*

Proof. As proof, we present an algorithm that generate

$$\mathcal{W}'_1 = \{x + e |\ x \textit{ be in condition 3 of event points definition}\}.$$

We consider each facility interval separately, because facility intervals don't effect each other. Consider a facility interval like FI such that its left endpoint is a P1's facility since if it is not then FI does not have $\mathcal{W}'_1$ points. Let this facility be f and $v_1, \ldots, v_m$ be the users in FI. We find points in set $FI \cap \mathcal{W}'_1$. We define RE to be right endpoint of FI.

Using Algorithm 1 we can find the set $\mathcal{W}'$. Now, we prove correctness of this algorithm. It is obvious that N_is are always equal to cov_{v_i} when we move across s_ks. Thus, based on Lemma 4 maximum of N_is is cov. Therefore, we always have value of cov, and whenever it changes, we have a $\mathcal{W}'$ point. About the

Algorithm 1 Make $\mathcal{W}'_1$

Input: points $f, RE, v_1, \ldots, v_m$
Output: set $\mathcal{W}'_1$

Let $S = \{max(f + 2(v_j - v_i), v_j) | 1 \leqslant i \leqslant j \leqslant m\}$.
Sort S.
Set $N_i = 0$ for $i = 1$ to m.
Set $max = N_1$.
for $k = 1$ to $sizeOf(S)$ **do**
  Let $s_k \in S$ correspond to i, j of S's definition.
  Increment N_i.
  if $N_i > max$ **then**
    Update max to N_i.
    Add s_k to $\mathcal{W}'_1$.
return $\mathcal{W}'_1$.

time complexity of the algorithm, S has $O(m^2)$ members and sorting of it take $O(m^2 \log m)$. For-loop repeats $O(m^2)$ time and each iteration of it takes $O(1)$. Therefore, total time complexity of the algorithm is $O(m^2 \log m)$.

We must save the values of $\operatorname{cov}(s, w'), w' \in \mathcal{W}'_1$ for later uses when we want to find optimum placement of last facilities of P1. We define $\mathcal{W}'_2$ to be:

$$\mathcal{W}'_2 = \{x - e | \ x \ \textit{be in condition 4 of event points definition}\}.$$

$\mathcal{W}'_2$ members can be compute similarly.

Based on above sayings it is obvious that $\mathcal{W}' = \mathcal{W}'_1 \cup \mathcal{W}'_2$ can be computed over all facility intervals in time $O(n^2 \log n)$.

After obtaining set $\mathcal{W}'$ we can make $\mathcal{W}$ by using e value. Assuming we have e value and set $\mathcal{W}$ we state a theorem for finding optimal placement of P1's facilities but before that we have the following lemma.

Lemma 6. *Let x and y be two points such that $x < y$ and there is not any facility in interval (x, y). If $v_1, \ldots, v_m$ is users in interval $(x, \frac{x+y}{2})$ then*

$$\operatorname{cov}(x, y) = max(N(\frac{x+y}{2}, y), \max_{u \in \{v_1, \ldots, v_m\}} cov_u(x, y))$$

.

Proof. Proof is similar to Lemma 4.

Theorem 8. *There is an algorithm that can find optimal placement of P1's last move in $O(n^2 \log n)$ time.*

Proof. We prove the theorem by dividing it into two cases. The first one restricts P1 to place his facilities in different facility intervals and the second one restricts him to place them in the same facility interval. According to Lemma 6 it is sufficient to examine $\mathcal{E}$ members in each case.

The first case is quite straightforward.

Consider every pair $x, y \in \mathcal{E}$ such that x and y belong to different facility intervals. We compute the maximum payoff of P2 by simulating its optimal placement if P1 places f_3 and f_4 on points x and y. This can be done in $O(1)$ for each x and y because we have saved all the required values in the course of computing $\mathcal{D}$ and $\mathcal{W}$. By Computing the payoff for every pair $x, y \in \mathcal{E}$, Theorem 6 guarantees that we can find the placement that maximizes the P1's payoff in this case. Time complexity of this case is $O(n^2)$.

The second case is a bit more complex. When x and y are in the same facility interval, say FI, we cannot simulate P2's next movement as easily as we did before. That is because we don't have $\text{cov}(x, y)$ stored, so we need to compute it every time when a facility moves. To obviate this, using a data structure which is a tree that is constructed over a given array of integers with size n somehow, the maximum element of each subarray can be find in $O(\log n)$. Such a tree is called **max tree**. We know that max tree can be constructed in $O(n)$ for an array with size n. We also know that if we change a leaf of this tree we can update it in time $O(\log n)$. Let $FI \cap \mathcal{E} = \{e_1, \ldots, e_m\}$ and $e_1 < \ldots < e_m$. We define RE and LE to be right endpoint and left endpoint of FI, respectively. Based on Theorem 6 examining $\{e_1, \ldots, e_m\}$ is sufficient. Therefore, it is obvious that we can find optimum placement of P1 in this case with considering all pairs of $\{e_1, \ldots, e_m\}$ and computing optimal placement of P2 using Theorem 4, but the time complexity of that will be $O(n^3)$. Although, Algorithm 2 can do this in $O(n^2 logn)$. In this algorithm we use Lemma 6 to find $\text{cov}(x, y)$.

Correctness of Algorithm 2 is trivial according to Theorem 4, because we compute all the listed numbers of Theorem 4 for all pairs of a sufficient set and finding the optimal placement of P2 for each pair. Note that in line 4, all leaves of the tree will be zero. Also each leaf will be updated at most m times. As the max tree has m leaves, it will be updated at most m^2 times during the Algorithm 2. Hence, runtime of max tree updates is $O(n^2 \log n)$.

Also payoff computation in step 9 is $O(logn)$ in each iteration and step 10 can be done in $O(logn)$ using binary search. Thus, the complexity of algorithm is $O(n^2 \log n)$. Hence, we can compute the optimal placements for all facility intervals in $O(n^2 \log n)$.

Eventually, we need to find all strategies from case one and two and choose the placement with maximum payoff for P1.

Now, the only remaining thing is computing e value. First and second property of is easy to satisfy. It is obvious that if e is less than difference of all consequent pairs in $U \cup \mathcal{D} \cup \mathcal{W}'$ then first and second property will satisfy. About the third property if e is less than difference of points in $U \cup \mathcal{D} \cup \mathcal{W}'$ and their nearest user from left an right, it is trivial that $N(x, y) = N(x - e, y + e)$. Therefore, we must only determine e to fit in third property for cov condition. First we present a lemma that we use to find e value.

Lemma 7. *Consider an interval like (LE, RE) such that $LE < RE$ and $LE, RE \in U \cup \mathcal{D} \cup \mathcal{W}'$. Let $\text{cov}(LE, RE) = m$ and $v_1, \ldots, v_m$ be m consequent users such*

Algorithm 2 Find optimal placement in FI with placing two facilities

Input: points $LE, RE, e_1, \ldots, e_m$

Output: optimal placement in FI with two facilities

Let $S = \{(x, y) | x, y \in \{e_1, \ldots, e_m\}, x < y\}$.
Sort S according to the absolute difference of elements of its members.
$\triangleright$Consider $s_1, s_2, \ldots, s_{m(m-1)/2}$ to be the increasing order of S
$s_1 = (a_1, b_1)$
Construct a max tree for array $[\mathrm{cov}(u, u + b_1 - a_1)], u \in U \cap FI$.
$\triangleright u$ is the label of the leaf.
for $i = 2, 3, \ldots, m(m-1)/2$ **do**
    Updata array $[\mathrm{cov}(u, u + b_{i-1} - a_{i-1})]$ to $[\mathrm{cov}(u, u + b_i - a_i)]$.
    Update max tree leaves according to $[\mathrm{cov}(u, u + b_i - a_i)]$.
    Update max tree according to new leaves.
    Find maximum of $cov_u(a_i, b_i)$, $u \in U \cap [a_i, \frac{a_i+b_i}{2}]$ via max tree.
$\triangleright cov_u(a_i, b_i)$ is the leaf max tree with label u.
    Compute $N(\frac{a_i+b_i}{2}, b_i)$ and find $\mathrm{cov}(a_i, b_i)$ via that and the result of previous step.
    Compute $N(a_i, b_i) - \mathrm{cov}(a_i, b_i)$.
    Compute the payoff of P1 in this case.
$\triangleright$We have all the listed number of Theorem 4, hence we can compute optimal payoff of P2 and then find payoff of P1.
    Update maximum payoff and corresponding placement.
return placement corresponded to maximum payoff.

that $v_m - v_1 \leqslant \frac{RE-LE}{2}$. *If* v_r *is the nearest user from right to* v_m *and* v_l *is the nearest user from left to* v_1*, then we have the following constraints on* e.

$$e < v_r - (\frac{RE - LE}{2} + v_1)$$

$$e < (v_m - \frac{RE - LE}{2} - v_l)$$

Proof. Consider $(LE - e, RE + e)interval$. According to Observation 2, $\mathrm{cov}(LE - e, RE + e) \geqslant N(v_1, v_1 + \frac{RE-LE}{2} + e)$. Therefore, if $\mathrm{cov}(LE - e, RE + e) = \mathrm{cov}(LE, RE)$, then $v_1 + \frac{RE-LE}{2} + e < v_r$ because if it does not, $\mathrm{cov}(LE - e, RE + e) \geqslant m + 1$. $v_1 + \frac{RE-LE}{2} + e < v_r$ conclude first constraint. Second constraint obtain from similar approach. Figure 2 give intuition about these constraitns.

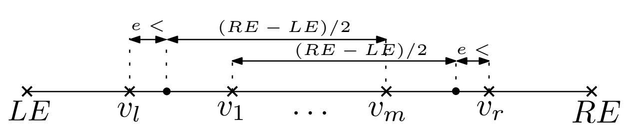

Fig. 2. e constraints

These constraints in Lemma 7 guarantee that $cov(LE-e, RE+e) = cov(LE, RE)$. Hence, if these constraints is satisfy for all $LE, RE \in U \cup \mathcal{D} \cup \mathcal{W}'$ then cov condition for third property of e will be satisfy.

Lemma 8. *Given points in $U \cup \mathcal{D} \cup \mathcal{W}'$, there is an algorithm that can find e value in $O(n^2 logn)$ time.*

Proof. e value has three property that said in Definition 4. First and second property are easy to satisfy by considering next and previous event points for each event point. Although, third property is a little tricky but can be found with an algorithm similar to Section 3. We examine all pairs of set $U \cup \mathcal{D} \cup \mathcal{W}'$ and find users that correspond to cov then we find constraints in Lemma 7. With those constraints and constraint for first and second property of e, we can find e value.

Lemma 9. *If P1 can place more than two facilities in last round. His placement can be computed in polynomial time.*

Proof. If we consider that P1 place k facilities in a facility interval then with considering cases that the number of facility in each facility interval is determined we can find the optimal solution. Therefore, let FI ba a facility interval that P1 wants to place k facilities in it. We can compute event points of FI and consider two of these event points like x and y, then we find solution for $k-2$ facilities in interval (x, y). With considering all cases we can find optimal solution.

4 Discrete blind voronoi game

In this section, we introduce a new version of voronoi game, called *blind voronoi game*. In blind version, all the situation is analogous to the discrete voronoi game, except the distribution of users is unknown to players. Notice that the distribution is fixed and not random. The only information which is known by a player is the number of users which are served by his facility in each round.

Deterministic placement is not a proper action in this game because it can cause zero payoff in general distribution. Instead of deterministic placement, a player must deem some candidate points for his placement and select one of these candidate points randomly to guarantee the minimum expected number of users. Therefore, the expected would be independent from the users' distribution. With this approach, players' target function is the minimum expected number of users which is desired to maximize by the corresponding player. Nonetheless, there is always a distribution of users such that the guaranteed minimum expected number of users for P1 is zero (as the last placement is located by P2, the distribution of users could be dense around that placement, so the expected number of P1's users could be zero in worst case). Hence, a natural optimization for P1 could be considered as minimizing the minimum expected number of the opponent's users (P2) instead.

Definition 6 (two-round one facility discrete blind voronoi game). *In this version, we have a given segment of line with unknown distribution of users, and each player has to locate a facility in each round. In each placement, each player knows the number of users which are served by him. The P1's goal is to minimize the minimum expected number of P2's users; conversely, maximizing the minimum expected number of P2's users is P2's aim.*

Notice that in the above definition, P1's goal is dependent on the minimum expected number of P2's users.

We say two facility are *adjacent* to each other if there is less than a constant number of users between them. This constant number is neglected in calculation of the expected number of users of players. Now we consider last move of P2. With respect to which facilities are adjacent, one of cases in Figure 3 can exist. There are some similar or symmetric cases that we omit. We specify the candidate points in Figure 3 with "cp". It is recognizable that all the candidate points that we determine is adjacent to P1's facilities. We show in the next lemma that it is a reasonable choice.

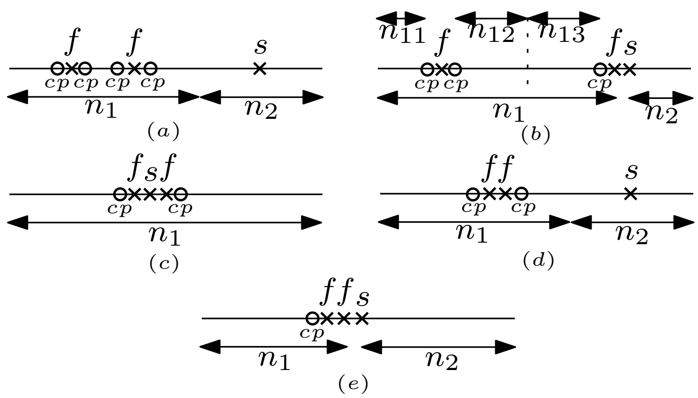

Fig. 3. blind voronoi game 1

Lemma 10. *Candidate points of P2 for his last move is adjacent to P1's facilities and in side of these facility that there is free of adjacent facilities. The probability of choosing these candidate points are equal.*

Proof. Let CS be a set of candidate points for P2's last move. As an important property, CS members must cover all the segment except some of it that we know have constant user in it like area between to adjacent facilities. If CS does not have this property then in a possible distribution, all the users can be in the area that CS members does not cover and minimum expected number of P2's users will be zero. It is obvious that in all cases of Figure 3, if one of determined candidate points does not belong to CS then there is an area between

that candidate points and its corresponding facility that has not been covered and can have more than constant number of users. Therefore, all the determined candidate points in Figure 3 are belong to CS.

Now, we must show that there is not any other point in CS. If there is any other point it should not be adjacent to any of P1's facility because all the adjacent points are in CS already. Let $x \in CS$ be a point that is not adjacent to any of P1's facility. Let $N_{rv}(S)$ be a random variable that shows the number of users that will serve by members of set S. Consider $\mathbb{E}(N_{rv}(CS \setminus x))$ and $\mathbb{E}(N_{rv}(CS))$ be the minimum expected number of users for candidate point sets $CS \setminus x$ and CS, respectively. Let the probability of choosing x in CS be P_x, then $\mathbb{E}(N_{rv}(CS)) = \mathbb{E}(N_{rv}(CS \setminus x)) * (1 - P_x) + N(x) * P_x$. Where $N(x)$ is the number of users that x covers in a possible distribution. In a possible distribution, the area that x covers can have zero users. Hence, $\mathbb{E}(N_{rv}(CS \setminus x)) > \mathbb{E}(N_{rv}(CS))$. This shows that, there is not any point in CS that is not adjacent to P1's facilities.

Now, we show that probability of CS members must be equal. Let probability of one of CS members be greater than other members. It is obvious that there is a possible distribution that number of users in the area of that member is zero. The minimum expected payoff in this situation is less than when the probabilities are equal. Hence, the probabilities of candidate points are equal.

In a formal manner, for example in case (b), our solution is like the below equation. This equation concludes that probabilities must be equal.

$$\begin{aligned} \max_{p_i} \min_{n_i} \ & p_1 n_{11} + p_2 n_{12} + p_3 n_{13} \\ \text{s.t. } & n_{11} + n_{12} + n_{13} = n - n_2 \\ & p_1 + p_2 + p_3 = 1 \\ & p_1, p_2, p_3, n_{11}, n_{12}, n_{13} \geqslant 0 \end{aligned} \tag{2}$$

Now we find the candidate points for last move of P2. In the next lemma we compute the expected payoff of P2 in each case of Figure 3.

Lemma 11. *Minimum expected payoff of P2 for his last move according to candidate points that determine in Figure 3 are* $\frac{1}{4}n$, $\frac{1}{3}n$, $\frac{1}{2}n$, $\frac{1}{2}n$ *and* n *for case (a), (b), (c), (d) and (e), respectively.*

Proof. According to Lemma 10, we know that the candidate points of each case are those that shown in Figure 3 and the probability of them are equal. In addition n_2 that is the number of users that P2 serve already can be zero and in a possible distribution $n_1 \simeq n$ can happen. Hence, if there is k candidate points then the minimum expected payoff of P2 will be $\frac{1}{k}n$. Therefore, the proposition of the lemma is correct.

According to the previous lemma, it is an unsuitable move if P1 place his second facility adjacent to his first facility or P2's first facility because in the case (c), (d) and (e) of Figure 3 that have such condition, the minimum expected payoff of P1 is greater than the other cases.

If P1 place his second facility non-adjacent to his and P2's first facility then only case (a) and (b) of Figure 3 can happen. In case (a) that P2's first facility is not adjacent to P1's facility, the minimum expected payoff of P2 is less than case (b) which P2 place his facility adjacent to P1's facility. Hence, P2 must place his first facility adjacent to P1's first facility.

In the next lemma we consider P1's last move with this assumption that P2's first facility is next to P1's first facility. Two cases might be happened that specified in Figure 4. Note that after P2 place his first facility, P1 will know amount of n_1 and n_2.

Lemma 12. *P1 must choose case (a) in Figure 4.*

Proof. The candidate points in two cases (a) and (b) of Figure 4 is like Figure 3 and the probability of them are equal. The only thing that we must show is that minimum expexted payoff of P2 in case (a) is less than case (b).

First consider case (a). It is obvious that in a possible distribution the number of users that facility s serve can be zero. Therefore by choosing one of points CP_1, CP_2 or CP_3, minimum expected payoff of P2 will be $\frac{1}{3}n$.

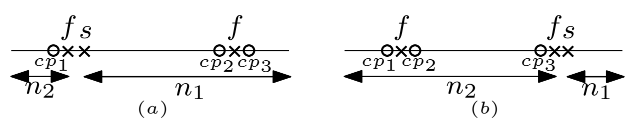

Fig. 4. blind voronoi game 2

In case (b), P2 already have all the users in right of the s which the number of them is n_1. Hence, by choosing one of points CP_1, CP_2 or CP_3, minimum expected payoff of P2 will be $\frac{1}{3}n_2 + n_1$. As we know $n_1 + n_2 \simeq n$, therefore $\frac{1}{3}n_2 + n_1 \simeq \frac{1}{3}n + \frac{2}{3}n_1$.

According to the above sayings the proposition of the lemma is trivial.

According to Lemma 12, P1 must select an arbitrary point in the desirable interval to make a situation like case (a).
The only moves that remained are first move of the players. We consider first move of P2 in the next lemma.

Lemma 13. *P2 must place his first facility on one of candidate point that determine in Figure 5 with equal probability.*

Proof. Proof of this lemma is similar to Lemma 10.

About the first move of game that is placement of first facility of P1, it is trivial that he must place his facility on an arbitrary point, because he has no information. In addition, according to lemmas of this section, place of P1's first facility does not have any impact on the minimum expected payoff of P2.

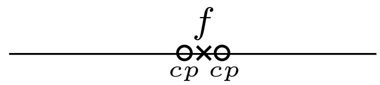

Fig. 5. blind voronoi game 3

5 Conclusion

In this paper we defined and found a set of placement for last move of every player in the discrete multi-facility Voronoi game. An straightforward algorithm for finding the best strategy of both players is devised by examining these placements and finding the placement that gives the player the maximum payoff. We also showed that this game doesn't have a winning strategy and there exists a case that neither of the players can win. At last we explained how we can generalize our method to solve the game for the case where m facilities are placed on the line in every round. As a future work we suggest working on games with more rounds.

References

1. Hee-Kap Ahn, Siu-Wing Cheng, Otfried Cheong, Mordecai Golin, and René van Oostrum. Competitive facility location: the voronoi game. *Theoretical Computer Science*, 310(1):457–467, 2004.
2. Sayan Bandyapadhyay, Aritra Banik, Sandip Das, and Hirak Sarkar. Voronoi game on graphs. *Theor. Comput. Sci.*, 562:270–282, 2015.
3. Aritra Banik, Bhaswar B. Bhattacharya, and Sandip Das. Optimal strategies for the one-round discrete voronoi game on a line. *J. Comb. Optim.*, 26(4):655–669, 2013.
4. Aritra Banik, Bhaswar B. Bhattacharya, Sandip Das, and Sreeja Das. Two-round discrete voronoi game along a line. In *Frontiers in Algorithmics* and *Algorithmic Aspects in Information and Management, Third Joint International Conference, FAW-AAIM 2013, Dalian, China, June 26-28, 2013. Proceedings*, pages 210–220, 2013.
5. Aritra Banik, Bhaswar B Bhattacharya, Sandip Das, and Satyaki Mukherjee. One-round discrete voronoi game in r2 in presence of existing facilities. CCCG, 2013.
6. Aritra Banik, Sandip Das, Anil Maheshwari, and Michiel H. M. Smid. The discrete voronoi game in a simple polygon. In *Computing and Combinatorics, 19th International Conference, COCOON 2013, Hangzhou, China, June 21-23, 2013. Proceedings*, pages 197–207, 2013.
7. Bhaswar B. Bhattacharya and Subhas C. Nandy. New variations of the maximum coverage facility location problem. *European Journal of Operational Research*, 224(3):477–485, 2013.
8. Sergio Cabello, José Miguel Díaz-Báñez, Stefan Langerman, Carlos Seara, and Inmaculada Ventura. Facility location problems in the plane based on reverse nearest neighbor queries. *European Journal of Operational Research*, 202(1):99–106, 2010.

9. Otfried Cheong, Sariel Har-Peled, Nathan Linial, and Jirí Matousek. The one-round voronoi game. In *Proceedings of the 18th Annual Symposium on Computational Geometry, Barcelona, Spain, June 5-7, 2002*, pages 97–101, 2002.
10. Christoph Dürr and Nguyen Kim Thang. Nash equilibria in voronoi games on graphs. In *Algorithms - ESA 2007, 15th Annual European Symposium, Eilat, Israel, October 8-10, 2007, Proceedings*, pages 17–28, 2007.
11. Sándor P. Fekete and Henk Meijer. The one-round voronoi game replayed. *Comput. Geom.*, 30(2):81–94, 2005.
12. Sachio Teramoto, Erik D. Demaine, and Ryuhei Uehara. The voronoi game on graphs and its complexity. *J. Graph Algorithms Appl.*, 15(4):485–501, 2011.

A Proofs

A.1 Proof of Theorem 6

We show that we can replace each two candidate points for P1's facilities, like x and y, with two points in $\mathcal{E}$ such that the optimum payoff of P2 reduces or remains unchanged. To prove this, we show that current payoff of P2 and each numbers that listed in Theorem 4 reduces or remains unchanged. Let x be an arbitrary point, served as P1's candidate facility, in an arbitrary location inside a facility interval (FI). Our goal is to show that we can move x to a member $\mathcal{E}$ such that the optimum payoff of P2 reduces or remains unchanged. We provide two general cases. In each case, there is an interval which the starting point is a facility which is owned by P1 or P2. Note that there is not any facility between the starting point and x. x is on the right side of mentioned facility, and we do not consider x to be on the left side of the facility due to similarity. Based on the owner of the starting point, these two general cases can be obtained:

Case 1 In this case, the starting point of the interval is a facility of P2(s in Figure 6). By selecting nearest point to x from the set of event points, two sub-cases will arise.

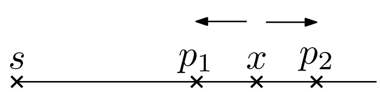

Fig. 6. Case 1

Case 1-1 In this case, the nearest event point is on the left side of x (p_1 in Figure 6). This point can be a member of U or $\mathcal{D}$. In both situations, $N(s,(x+s)/2) = N(s,(p_1+s)/2)$ because there is no member of D between x and p_1. Thus, the P2's optimum payoff corresponding to s, remains unchanged. If $p_1 \in U$, then $N(s,p_1) = N(s,x) - 1$, and if $p_1 \in \mathcal{D}$, $N(s,p_1) = N(s,x)$. This fact clearly demonstrate that by substituting p_1 for x, P2's optimum payoff does not increase.

Case 1-2 In this case, the nearest event point is on the right side of x (p_2 in Figure 6). The proof of this case is similar to previous one.

Case 2 In this case, the starting point of the interval is a facility of P1 (f in Figure 7). Based on the location of nearest event point, similar to Case 1, we have two sub-cases.

Case 2-1 In this case, the nearest event point is on the left side of x (p_1 in Figure 7). This point can be a member of U or $\mathcal{W}$. In both situations, because $(f,p_1) \subset (f,x)$, we have $\mathrm{cov}(f,p_1) \leqslant \mathrm{cov}(f,x)$ from Lemma 3. Also, $N(f,p_1) \leqslant N(f,x)$ in both cases. Therefore, P2's optimum payoff does not increase.

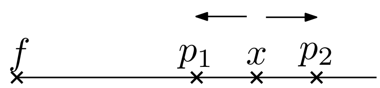

Fig. 7. Case 2

Case 2-2 In this case, the nearest event point is on the right side of x (p_2 in Figure 7). Exactly same as above case, $p_2 \in U$ or $p_2 \in \mathcal{W}$. In both situations, we can be sure that $cov(f, p_2) = cov(f, x)$ and $N(f, p_2) = N(f, x)$ if the interval (x, p_2) be free of any member of $\mathcal{W}'$; hence, P2's optimum payoff does not increase. Now, if there is a member of $\mathcal{W}'$ inside (x, p_2), $\text{cov}(f, p_2) > \text{cov}(f, x)$, and this shows an increase in listed numbers of Theorem 4 which is not acceptable. In such a situation, let $w' \in \mathcal{W}'$ be inside aforementioned interval, and the respective point on set $\mathcal{W}$ be w. By the definition of e the interval (w, w') is free of any point from $\mathcal{W}'$ or $\mathcal{E}$. Therefore, $\text{cov}(f, w) = \text{cov}(f, x)$ and $N(f, w) = N(f, x)$. Thus, we will replace x with w, rather than the nearest event point.

In second part of the proof, we consider placement format of P1's facilities of this round (f_3, f_4).

Case 1 In this case, f_3 and f_4 are in two different FI in an arbitrary location. We replace each facility with nearest event point. Due to the fact that there is P1 or P2's facilities at the end of each FI, these nearest event points can be considered as p_1 or p_2 in Figures 6 and 7. Based on this view point, we can have cases 1-1, 1-2, 2-1, or 2-2. If cases 1-1, 1-2, or 2-1 have been happening, the P2's optimum payoff does not increase. If our arrangement reaches to case 2-2, it is possible to have two separately cases. First, there is not any $\mathcal{W}'$'s member between x and nearest event point. In this case, based on the above mentioned claims, the P2's optimum payoff will decrease or remains unchanged. On the other hand, if there is a member of $\mathcal{W}'$ within the interval of x and nearest event point, as suggested before, we place P1's facility on the respective $w \in \mathcal{W}$ of $w' \in \mathcal{W}'$. Due to the absence of any event points or $\mathcal{W}'$'s member inside interval between w and w', the P2's optimum payoff remains unchanged.

Case 2 In this case, f_3 and f_4 are in a common FI in an arbitrary location. In this manner, we keep f_3 and f_4 approaching to each other until reaching two event points. We act like before according to presence or absence of any member of $\mathcal{W}'$ in the intervals between f_3 and f_4 and their corresponding nearest event point. According to the definition of e it is obvious that the optimum payoff of P2 will not increase. Look at Figure 8 for illustration. For first and third intervals, the arrangement is same as Case 1. By the definition of e, it is obvious that cover and number of users in the second interval will not increase.

So, in all situations the P2's optimum payoff does not increase.

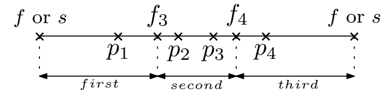

Fig. 8. Case 2

A.2 Lack of Winning Strategy in Discrete Case

B And I also came across this

But I need to put this in an appendix so that my paper is not too long.